

Service Level Expectation

"A Expectativa de Nível de Serviço (Service Level Expectation), prevê quanto tempo um item deve fluir do início ao fim do fluxo de trabalho do Time Scrum. O Time Scrum utiliza esse SLE para encontrar problemas no seu fluxo ativo, inspecionando-os e adaptando-os em casos de queda abaixo de suas expectativas. O SLE em si, possui duas partes: um período de dias corridos e uma probabilidade associada a esse período (e.g., 85% dos itens de trabalho devem estar finalizados em 8 dias ou menos). O SLE deve ser baseado no cycle time histórico do Time Scrum e, uma vez calculado, o Time Scrum deve deixá-lo transparente. Caso não haja um histórico de cycle time, o Time Scrum deverá fazer sua melhor estimativa e então substituí-la quando houver dados históricos suficientes para realizar um cálculo mais apropriado do SLE."

Conteúdo retirado do Kanban Guide for Scrum (http://abre.ai/kanban_guide_for_scrum)

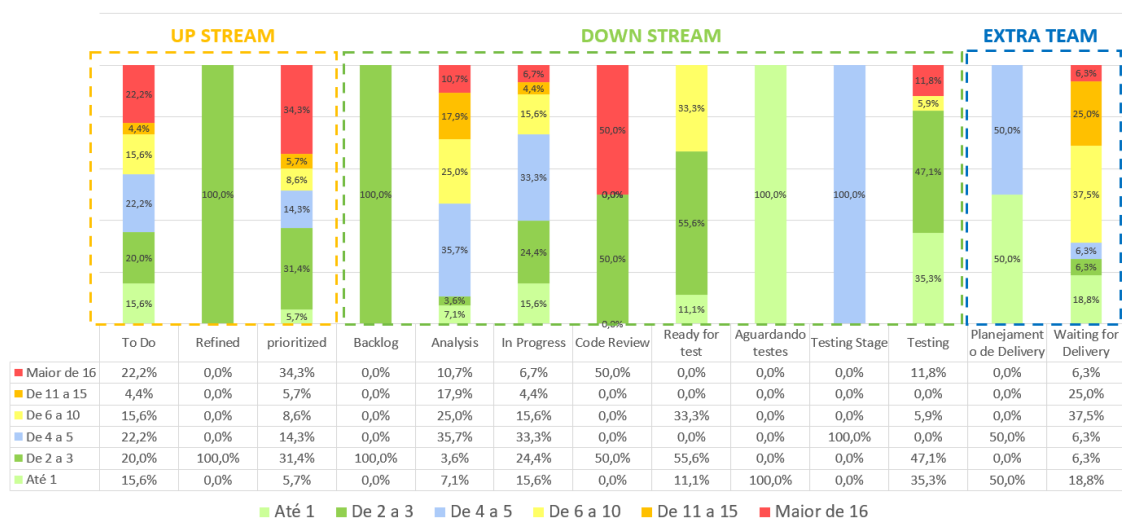
Por definição, o SLE serve para aprimorar a previsibilidade de prazos e identificar falhas no fluxo (completo da Sprint) de itens que estejam ultrapassando a previsão.

Vamos agora entender uma definição que tem atrapalhado o entendimento de muitos. Quem fez KMP I e KMP II entendem que CYCLE TIME é o período em que um item se manteve dentro de uma faixa do Kanban (se manteve em um status no Jira) e LEAD TIME é o tempo do fluxo completo, desde a criação até ficar DONE. Vocês estão corretos.

Porém, é importante ressaltar, que o SLE considera o período completo, mas no Guide trata como CYCLE TIME. Isso tem gerado uma série de dúvidas e para quem procura certificação PSK, o termo correto é CYCLE TIME mesmo, pois considera o período que compõe a Sprint e não desde o momento em que o item é criado até ficar DONE.

Então, SLE trabalha com o que muitos entendem como LEAD TIME e em minha proposta, eu adaptei as visibilidades e considerei o CYCLE TIME de cada uma das faixas, para conseguir uma visão de cada etapa do meu Fluxo e poder melhorá-lo.

ANÁLISE POR FAIXA



COMO EXTRAIR OS DADOS - JIRA SERVER

Para os usuários do Jira Server, existe uma API que pode ser acessada: <https://developer.atlassian.com/server/jira/platform/rest-apis/>

Podemos utilizar um programa feito em Python que se conecta à API do Jira: <https://jira.readthedocs.io/en/master/>

Será necessário a instalação do Python em suas máquinas (<https://dicasdepython.com.br/como-instalar-o-python-no-windows-10/>) e também do Visual Studio. Quem não tiver deverá abrir um chamado para a instalação. Ao rodar, algumas bibliotecas serão necessárias (<https://computadorcomwindows.com/2018/01/19/tutorial-como-instalar-uma-biblioteca-python-no-computador/>)

O PYTHON UTILIZA IDENTIFICAÇÃO PARA CONTROLAR SUA ESTRUTURA, CUIDADO COM O CTRL+V POIS COSTUMA PULAR LINHAS E PERDER A IDENTIFICAÇÃO

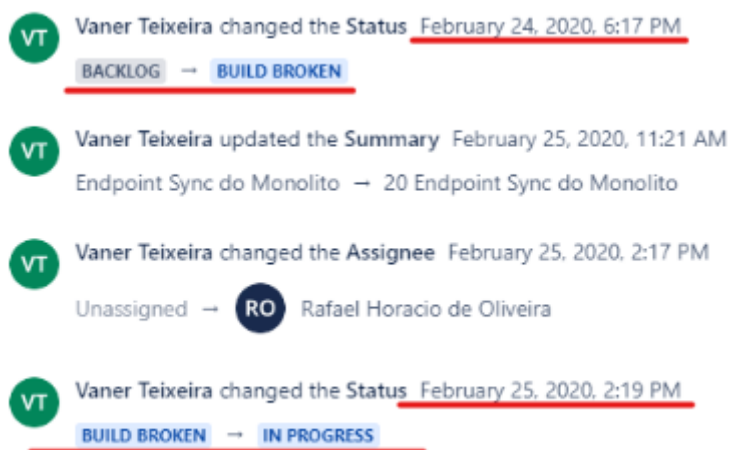
Antes de rodar é necessário inserir os seus dados de login e projeto. Funciona somente para o Jira Server.

Toda as configurações são feitas no arquivo ConsultaAPI.py, lá vc também configura o seu JQL com as informações necessárias. O Arquivo Métricas de Eficiência.xlsx já é abastecido automaticamente, caso mudem algum parâmetro, deverão ser inseridos ou retirados os campos.

COMO EXTRAIR OS DADOS - JIRA CLOUD

É meus amigos, está opção não é indolor. Infelizmente, ainda não possuímos uma API para o Jira Cloud, então a extração é na mão.

Tenho montado uma planilha e inserido os dados a partir do History de cada um dos Itens. Caso o time desenvolva Sub-Tasks, eu acesso cada uma e vou anotando as datas das mudanças de status.



Coletou e inseriu a informação na planilha.

	CREATE		SPRINT		IN PROGRESS		CODE REVIEW
	BACKLOG		BUILD BROKEN		IN PROGRESS		BUILDING
NOV-62	10/02/20	0	10/02/20	3	13/02/20	8	
NOV-33	29/01/20	12	10/02/20	2	12/02/20	1	13/02/20
NOV-45	29/01/20	12	10/02/20	7	17/02/20	1	18/02/20
NOV-46	29/01/20	12	10/02/20	8	18/02/20	1	
NOV-47	29/01/20	12	10/02/20	3	13/02/20	6	19/02/20
NOV-48	29/01/20	12	10/02/20	9	19/02/20	2	21/02/20
NOV-67	17/02/20	0	17/02/20	0	17/02/20	1	
NOV-69	17/02/20	0	17/02/20	0	17/02/20	1	
NOV-24	10/02/20	0	10/02/20	1	11/02/20	1	12/02/20
NOV-65	10/02/20	0	10/02/20	1	11/02/20	1	12/02/20
NOV-51	29/01/20	12	10/02/20	2	12/02/20	2	14/02/20
NOV-52	29/01/20	12	10/02/20	8	18/02/20	3	
NOV-53	29/01/20	12	10/02/20	9	19/02/20	0	19/02/20
NOV-54	29/01/20	12	10/02/20	4	14/02/20	3	17/02/20
NOV-55	29/01/20	12	10/02/20	8	18/02/20	2	20/02/20
NOV-56	29/01/20	12	10/02/20	9	19/02/20	1	20/02/20
NOV-57	29/01/20	12	10/02/20	7	17/02/20	1	18/02/20
NOV-58	29/01/20	12	10/02/20	7	17/02/20	1	18/02/20
NOV-63	10/02/20	0	10/02/20	1	11/02/20	7	18/02/20
NOV-70	19/02/20	0	19/02/20	0	19/02/20	0	

COMO GERAR OS GRÁFICOS

Após extrair os dados, precisamos calcular a quantidade de dias de cada Cycle Time, para isso, formate a coluna como NÚMERO sem casas decimais e faça a subtração de uma coluna de data por outra, no exemplo abaixo, a célula C3 é =D3 - B3

A	B	C	D	E	F
	CREATE		SPRINT		IN PROGRESS
	BACKLOG		BUILD BROKEN		IN PROGRESS
NOV-62	10/02/20	0	10/02/20	3	13/02/20
NOV-33	29/01/20	12	10/02/20	2	12/02/20
NOV-45	29/01/20	12	10/02/20	7	17/02/20
NOV-46	29/01/20	12	10/02/20	8	18/02/20
NOV-47	29/01/20	12	10/02/20	3	13/02/20
NOV-48	29/01/20	12	10/02/20	9	19/02/20

Feito isso para cada um dos Cycle Times, precisamos criar os nossos períodos e acrescentar as contagens. Fiz da seguinte forma:

	Até 1 dia	de 2 a 3	de 4 a 5	5 a 10	+ de 10
BACKLOG	17	0	0	0	16
SPRINT	6	4	3	11	1
IN PROGRESS	12	7	1	5	0
CODE REVIEW	10	2	2	1	0
TESTING QA	8	4	3	0	0

Para calcular os itens de até um dia, utilizei a fórmula CONT.A, adequando para as respectivas colunas de cada status, a fórmula é essa: =CONT.A(\$C:\$C;"<=1")

Para as demais, como o intervalo inclui um período de dois dias, utilizei a fórmula CONT.B, adequando para as respectivas colunas de cada status, a fórmula é essa: =CONT.B(\$C:\$C;">1";\$C:\$C;"<4")

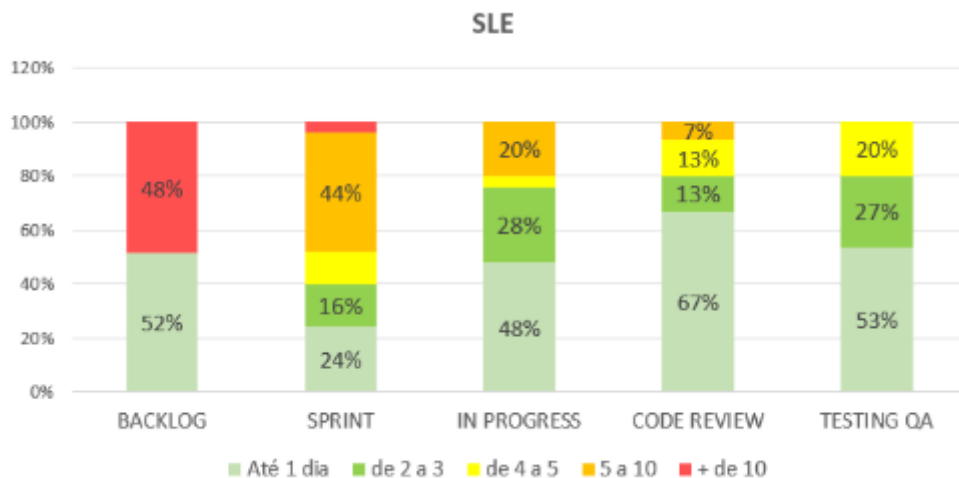
Agora precisamos totalizar os itens que passaram por cada uma das faixas e depois calcularmos as porcentagens. Insira o total de cada faixa.

	BACKLOG	SPRINT	IN PROGRESS	CODE REVIEW	TESTING QA
Até 1 dia	17	6	12	10	8
de 2 a 3	0	4	7	2	4
de 4 a 5	0	3	1	2	3
5 a 10	0	11	5	1	0
+ de 10	16	1	0	0	0
	33	25	25	15	15

Tendo o total de cada faixa, criei um novo conjunto de dados e comecei a calcular as porcentagens, basta dividir o conteúdo de cada célula pelo total de cada coluna. Neste novo conjunto, formate as células para porcentagem sem casa decimal.

	BACKLOG	SPRINT	IN PROGRESS	CODE REVIEW	TESTING QA
Até 1 dia	52%	24%	48%	67%	53%
de 2 a 3	0%	16%	28%	13%	27%
de 4 a 5	0%	12%	4%	13%	20%
5 a 10	0%	44%	20%	7%	0%
+ de 10	48%	4%	0%	0%	0%

Agora é só gerar o gráfico.



O ideal é manter o conjunto de dados dos últimos 120 dias, pois assim, desvios e melhorias poderão ser detectados com facilidade.

Uma outra forma de realizar o SLE, é de fato pelo LEAD TIME e isso você poderia fazer pelo LEADTIME total e o da Sprint. Precisa da Data de Criação do Item, Data de Entrada na Sprint ou de Início do Desenvolvimento e a Data de Conclusão.

ITEM	DATA CREATE	DATA SPRINT	DONE	LEADTIME GERAL	LEADTIME SPRINT
NOV-54	10/02/20	10/02/20	20/02/20	10	10
NOV-55	29/01/20	10/02/20	21/02/20	23	11
NOV-56	29/01/20	24/02/20	25/02/20	27	1
NOV-57	29/01/20	24/02/20	27/02/20	29	3
NOV-58	29/01/20	24/02/20	25/02/20	27	1
NOV-63	29/01/20	24/02/20	26/02/20	28	2
NOV-70	17/02/20	24/02/20	28/02/20	11	4
NOV-49	17/02/20	24/02/20	05/03/20	17	10
NOV-50	10/02/20	02/03/20	06/03/20	25	4
NOV-54	10/02/20	05/03/20	08/03/20	27	3
NOV-73	29/01/20	24/02/20	28/02/20	30	4

Da mesma forma, podemos gerar as faixas e gerar os cálculos com visões distintas para os dois LEADTIMES.

